Professional and Scientific Staff Management, Inc.
Temporary Staff Fulfillment System
Deployment Proposal
2016-12-07

Rob Sanchez and Juan Carcamo
Knowledge Machines Analysis & Design

CIS 641, Fall 2016
Schymik

Contents

1. Introduction

The purpose of this document is to present a deployment proposal for a Temporary Staffing Fulfillment System (TSFS) at Professional and Scientific Staff Management, Inc. of Toledo, OH. PSSM is a medium-sized, niche-market staffing agency operating in the Midwestern United States. It specializes in high-skilled technical and scientific temporary staffing placements and has experienced an unexpected surge in demand for suitable temporary staffing candidates from companies throughout the region in the last three years.

PSSM's current placement and arrangements system, such as it is, hasn't been able to keep up with the agency's growth and suffers from significant performance and usability problems. As a result, leadership has decided to replace it. They reached out to our firm, *Knowledge Machines Analysis & Design*, for a re-implementation of their solution.

PSSM's core competency is negotiating contracts and maintaining relationships between their counter-parties. They employ a minimal IT team with no plans to hire new staff. Their current staffing system consists of a set of related spreadsheets stored on the company's shared network drive. This arrangement does not support a high level of concurrency and suffers from a high rate of errors. It also doesn't protect client or candidate staff personal data to the satisfaction of the PSSM legal department.

PSSM are looking for a hosted, cloud-based web application to access from their offices in downtown Toledo, with physical data management to be handled offsite by a third party. Basic web application configuration access should be available to PSSM IT staff, but code development and maintenance is to be managed by KMAD through a standard support contract.

2. Description

2.1 Requirements

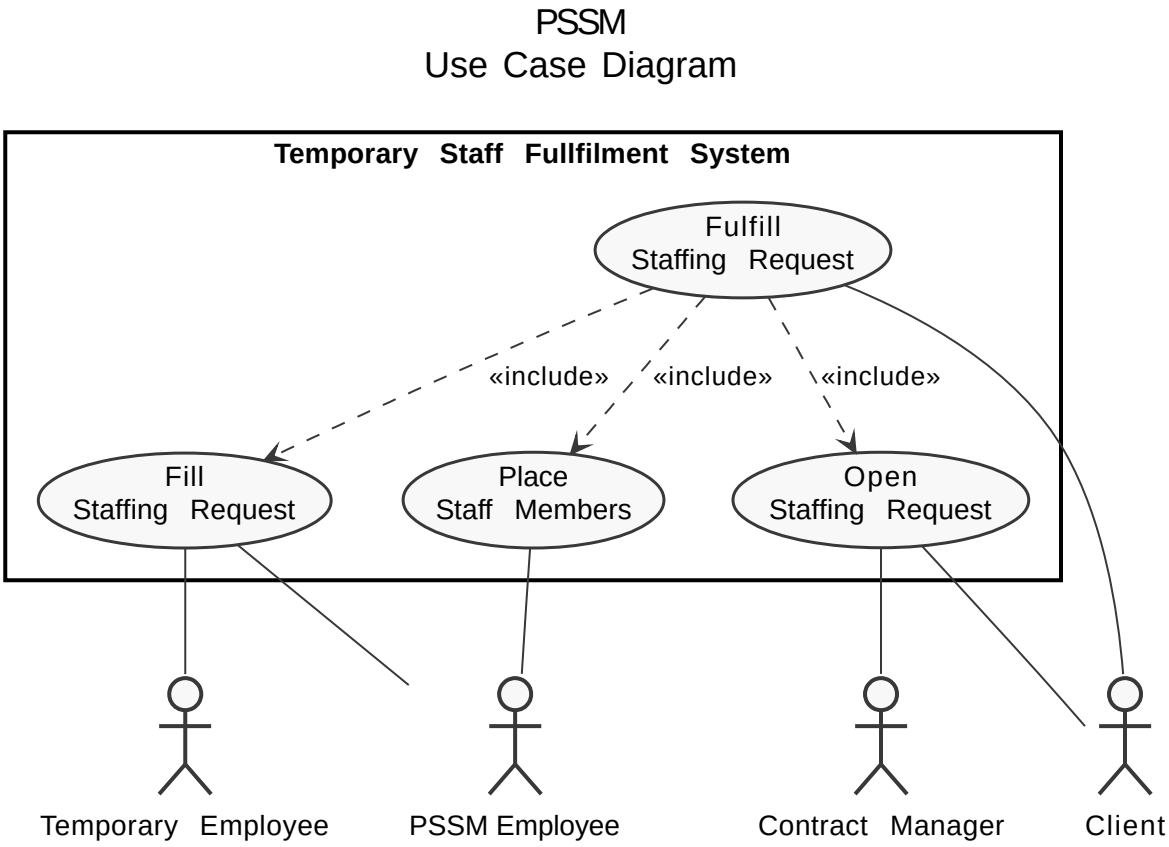
3. Effort

4. Analysis

4.1 Functional

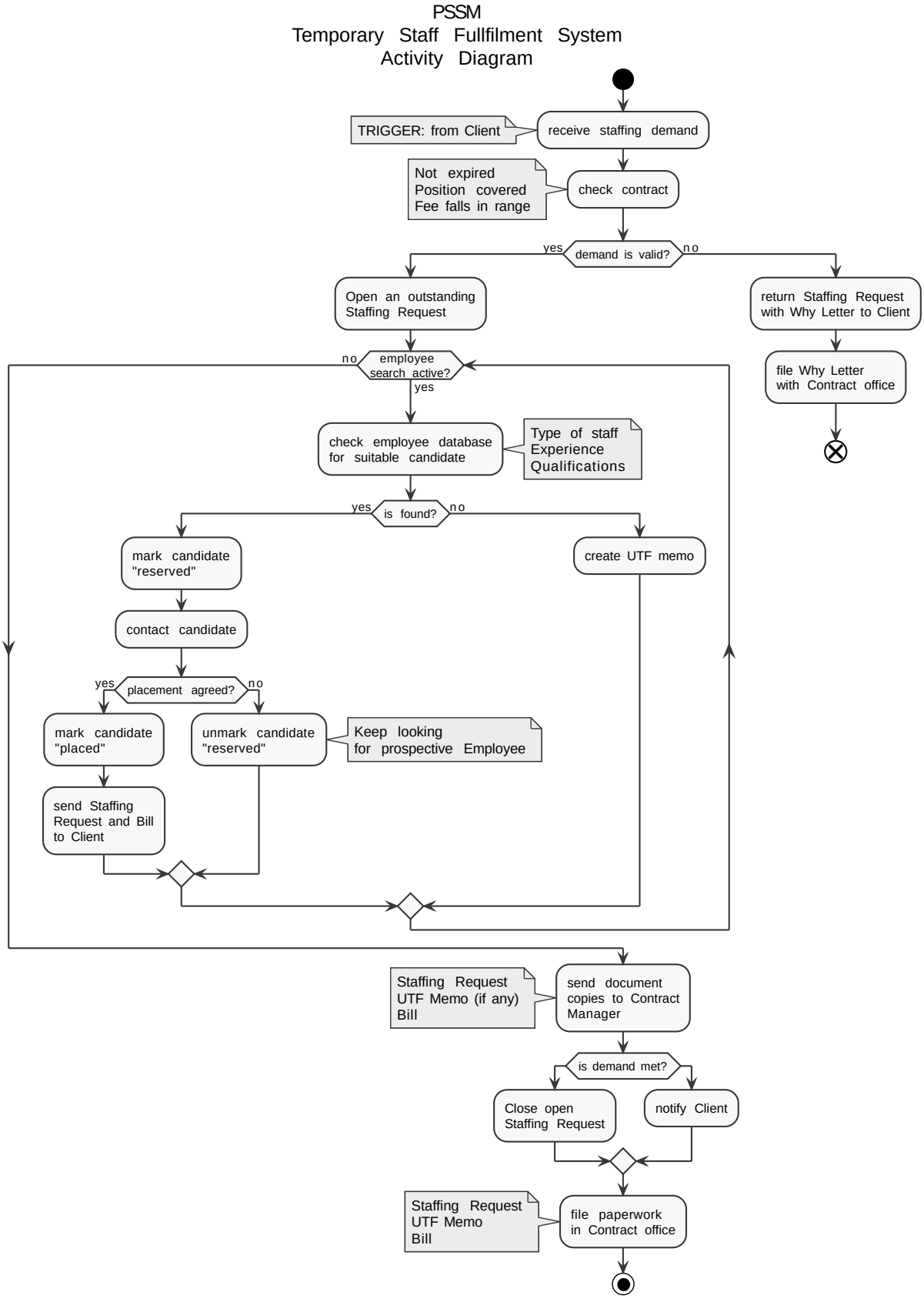
4.1.1 Use Case Diagram

The following Use Case Diagram depicts the main business processes and actors required by the Temporary Staffing System.



4.1.2 Activity Diagram

The following Activity Diagram



4.1.3 Use Case Descriptions

The following four Use Case Descriptions functionally document the business processes depicted in the Use Case Diagram and Activity Diagram presented in sections 4.1.1 and 4.1.2. These are used to build the structural models for the Staffing System.

Use Case Name:	Open Staffing Request (detailed)
Scenario:	Create a new outstanding Staffing Request.
Triggering event:	Client contacts PSSM and demands a new temporary staff arrangement.
Brief description:	PSSM receives a new demand for temporary staffing from the Client. PSSM's Contract Manager reviews the contract terms. If the new demand is not valid, the demand is returned to the Client with a Why Letter explaining. Otherwise, a new Staffing Request is opened in the Contract database and Placement is notified.
Actors:	Client, Contract Manager
Related use cases:	Place Staff Members, Fulfill Staffing Request
Stakeholders:	Contract Manager, Temporary Employee, Placement department
Pre-conditions:	A valid demand.
Post-conditions:	A new outstanding Staffing Request is created and the Placement department is notified.

Flow of activities:		
	Actor	System
	1. Client demands a new temporary staff arrangement 2. Contract manager reviews the client contract 3. Contract manager opens a new Staffing Request	2-1. Referencing the contract number on the new demand, the System pulls the Client contract details from the Contract database. 2-2. System creates a New Vehicle entry and saves associated details: serial number, description, make, model, features, etc. 3-1. System creates a new Staffing Request, marking it outstanding 3-2. System notifies Placement department of outstanding Staffing Request.

Exception conditions:	2-1. Demand is invalid, send Why Letter to Client explaining. File letter locally.

Use Case Name:	Place Staff Members (detailed)
Scenario:	Find suitable temporary staff to fill a client demand.
Triggering event:	Contract Manager opens a new outstanding Staffing Request, and the System notifies the Placement department.
Brief description:	Placement department checks position type, experience, and qualifications specified on the Staffing Request in the staff database.
Actors:	Placement department: PSSM Employee
Related use cases:	Open Staffing Request, Fill Staffing Request, Fulfill Staffing Request
Stakeholders:	Placement department, Contract Manager, Arrangements department, Temporary Employee
Pre-conditions:	A new outstanding Staffing Request must exist. A qualified Temporary Employee must be available.
Post-conditions:	A qualified Temporary Employee is marked as “reserved” and the Arrangements department is notified.

Flow of activities:		
	Actor	System
	1. A PSSM Employee from Placement checks the position type, experience, and qualifications specified in the Staffing Request against the database of available Temporary Employees. 2. The Temporary Employee is marked as “reserved”	1-1. System pulls available Temporary Employees matching the Staffing Request. 2-1. The System forwards the Staffing Request to the Arrangements department and notifies them that a possible match was found.

Exception conditions:	2-1. A qualified Temporary Employee is not available and an “unable-to-fill” Memo is explaining is attached to the Staffing Request.

Use Case Name:	Fill Staffing Request (detailed)
Scenario:	Arrange a Temporary Employee assignment for a Client.
Triggering event:	A possible qualified and available staff candidate is marked as “reserved” in the System, and Arrangements department is notified.
Brief description:	A potential temporary staff candidate is contacted, and after settling details and negotiating terms, the Temporary Employee is placed with the client and the Staffing Request is Closed.
Actors:	Arrangements department: PSSM Employee, Temporary Employee
Related use cases:	Place Staff Members, Fulfill Staffing Request
Stakeholders:	Client, Temporary Employee, Arrangements department
Pre-conditions:	A possible qualified and available temporary staff candidate has been marked as “reserved” in the System.
Post-conditions:	A Temporary Employee is marked as “placed” in the System. The open Staffing Request is closed. A copy of the original Staffing Request and a Bill is sent to the Client. The Staffing Request, any “unable-to-fill” memos and a copy of the Bill are sent to the Contract Manager.

Flow of activities:		
	Actor	System
	1. A PSSM Arrangements Employee contacts the temporary staff candidate. 2. The PSSM Employee settles any details and negotiates terms with the candidate. 3. The PSSM Employee places the Temporary Employee with the Client. 4. The PSSM Employee closes the open Staffing Request.	3-1. The System marks the Temporary Employee as “placed”. 4-1. The System closes the open Staffing Request. 4-2. Copy of the Staffing Request and a Bill are sent to the client. 4-3. The Staffing Request, a copy of the Bill, and any “unable-to-fill” Memo is sent to the Contract Manager.

Exception conditions:	3-1. Staffing Request not filled, client is notified. Staffing Request, Bill, and “unable-to-fill” Memo are filed in the contract office.

Use Case Name:	Fulfill Staffing Request (overview)
Scenario:	Open and fill a Staffing Request in order to place a Temporary Employee with a Client to meet a demand.
Triggering event:	A Client contacts PSSM to demand a new temporary staff arrangement.
Brief description:	The PSSM Contract Manager receives the Client demand and reviews the contract details. The Placement department finds a suitable staff member to meet the Client demand. Arrangements department negotiates the arrangement and places the employee with the Client.
Actors:	Client
Related use cases:	Open Staffing Request, Place Staff Members, Fill Staffing Request
Stakeholders:	Client
Pre-conditions:	Client and PSSM have an active contract in place.
Post-conditions:	A Temporary Employee is placed with the Client.

Flow of activities:		
	Actor	System
	1. Contract Manager opens a Staffing Request. 2. The Placement department identifies a suitable staffing candidate. 3. The Arrangements department negotiates with and places the Temporary Employee with the Client.	1. Placement department is notified. 2. Arrangements department is notified. 3. The Client is billed.

Exception conditions:	2-1. The Staffing Request cannot be fulfilled, the client is notified. The SR, Bill, and “unable-to-fill” Memo are filed in the Contract office.

4.2

5. Design

   The system will be named *TechStaff*.

6. User Interface

7. Physical Architecture

8. Testing

A System Test Plan will allow PSSM to gain confidence that KMAD listened to PSSM's needs and implemented the solution to meet them. Here we propose a Testing Plan with similar and interrelated phases: Requirements Testing, Documentation Testing, Alpha Testing and Beta Testing. We will provide a set of test cases and use them during all four phases.

8.1 Requirements Testing

Performed by KMAD, does it fulfill the business requirements?
When?

8.2 Documentation Testing

Performed by KMAD, does the documentation match the application?
When?

8.3 Alpha Testing

Performed by PSSM staff, with test data.
When?

8.4 Beta Testing

Performed by PSSM staff, with real data under close supervision by KMAD and PSSM IT.
When?

8.5 Test Cases

As discussed, the following test cases will be used in all four testing phases outlined above.

Test case #: 1	*Test Case Name:* Login/Logout/Reset Password
System: TechStaff	*Subsystem:* Security
Designed by: Larry David	*Design Date:* 2016-11-15
Executed by: Cosmo Kramer	*Execution Date:* 2016-11-30
Short Description: Test the option code field during sales Invoice data entry.	

Pre-conditions:

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1				
2				
3				
4	*Check post-condition 1.*			
5				
6				
7				
8	*Check post-condition 2.*			
9				
10				
11				
12	*Check post-condition 3.*			
13				
14				
15				
16	*Check post-condition 4.*			

Post-conditions:

1. The new TempEmployee is saved in the database with the entered data.
2. The is saved in the database with all dealer options removed.

3. The is saved in the database with the selected dealer option (typed input).

Test case #: 2	*Test Case Name:* CRUD Contract
System: TechStaff	*Subsystem:* Contracts
Designed by: Wade Wilson	*Design Date:* 2016-11-15
Executed by: Bruce Wayne	*Execution Date:* 2016-11-30

Short Description: Test the Creation, Reading, Updating, and Deleting of a client Contract

Pre-conditions:

The Customer has established a contract with a client.

The Customer has logged into the the TechStaff web application.

Step	*Action*	*Expected System Response*	*Pass/ Fail*	*Comment*
1	Test Create: Click the "Create Contract" button	The system presents a list of input fields. Fields surrounded by a red box are required.		
2	Fill in required Contract fields: ID, Expiration Date, Name, Address, Phone, Terms	The system enables the Save button at the bottom of the input fields		
3	Click Save.	The system presents a pop-up message "Contract #<ContractID>" has been saved. The list of existing contracts is displayed.		
4	*Check post-condition 1.*			
5	Test Read: Click the "Find Contract" button	The system presents a list of Contract search input fields: ID, Expiration Date, Name, Phone		
6	Type the desired Contract ID into the "ID" field. Click "Search"	The system presents a list of matching contracts.		
7	Select the desired Contract and click the "Open" button	The system displays the Contract details. The fields are read-only.		
8	*Repeat Test Read (5) for Expiration Date, Name, and Phone.*			
9	Test Update: Open a Contract as described in Test Read (5) above.	The system displays the Contract details. The fields are read-only.		
10	Click the "Edit" button	The Contract fields become editable,		

		except for the Contract ID field, which can't be changed.		
11	Enter a change for the Phone field. Click Save.	The system presents a pop-up message "Contract #<ContractID>" has been saved.		
12	*Check post-condition 2.*			
13	*Repeat Test Update (9) for Expiration Date, Name, and Phone.*			
14	Test Delete: Open a Contract as described in Test Read (5) above.	The system displays the Contract details. The fields are read-only.		
15	Click the "Edit" button	The Contract fields become editable, except for the Contract ID field, which can't be changed.		
16	Click the "Delete Contract" button	The system displays a pop-up message "Are you sure you want to delete Contract #<ContractID>? You cannot undo this action." Two buttons are presented below: "Cancel" and "Delete"		
17	Click Delete.	The system displays a message, "Contract #<ContractID> has been deleted."		
18	*Check post-condition 3.*			

Post-conditions:

1. The new Contract is saved in the database with the entered data.
2. The updated Contract is saved in the database with the new data.
3. The Contract has been removed from the database.

Test case #: 3 *Test Case Name:* CRUD TempEmployee

System: TechStaff *Subsystem:* Employee

Designed by: Barry Allen *Design Date:* 2016-11-15

Executed by: Doctor Strangelove *Execution Date:* 2016-11-30

Short Description: Test the Creation, Reading, Updating, and Deleting of a client Contract

Pre-conditions:

Step	*Action*	*Expected System Response*	*Pass/*	*Comment*

			Fail	
1				
2				
3				
4	*Check post-condition 1.*			
5				
6				
7				
8	*Check post-condition 2.*			
9				
10				
11				
12	*Check post-condition 3.*			
13				
14				
15				
16	*Check post-condition 4.*			

Post-conditions:

1. The new TempEmployee is saved in the database with the entered data.
2. The is saved in the database with all dealer options removed.
3. The is saved in the database with the selected dealer option (typed input).

Test case #: 4 *Test Case Name:* CRUD StaffingRequest

System: TechStaff *Subsystem:* Employee

Designed by: Larry David *Design Date:* 2016-11-15

Executed by: George Costanza *Execution Date:* 2016-11-30

Short Description: Test the Creation, Reading, Updating, and Deleting of a StaffingRequest

Pre-conditions:

Step	*Action*	*Expected System Response*	*Pass/*	*Comment*

			Fail	
1				
2				
3				
4	*Check post-condition 1.*			
5				
6				
7				
8	*Check post-condition 2.*			
9				
10				
11				
12	*Check post-condition 3.*			
13				
14				
15				
16	*Check post-condition 4.*			

Post-conditions:

1. The new TempEmployee is saved in the database with the entered data.
2. The is saved in the database with all dealer options removed.
3. The is saved in the database with the selected dealer option (typed input).

9. Change Management

10. Conclusion